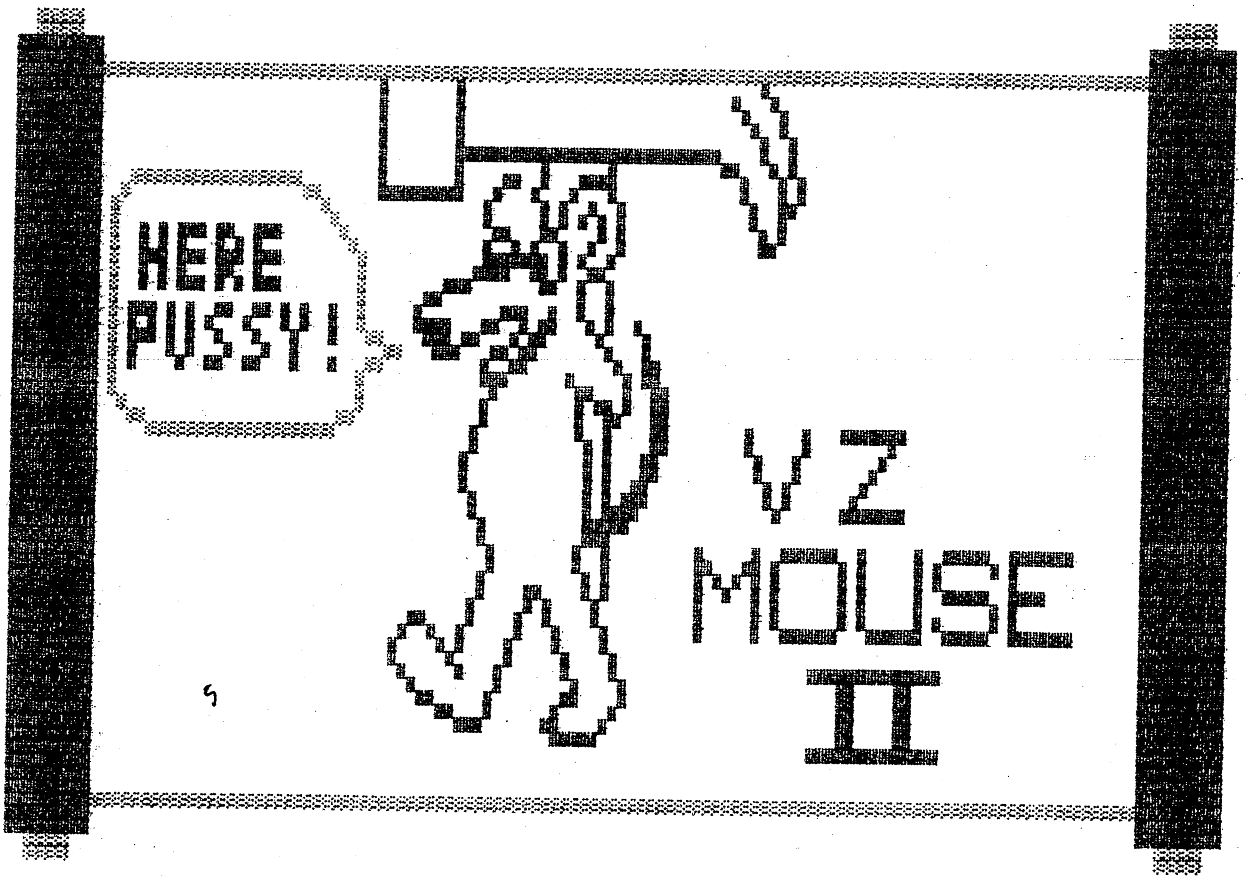
OCTOBER 1992 ISSUE # 40

HUNTER VALLEY VZ JOURNAL

VZ 200/300



PRODUCED BI-MONTHLY BY H.V.VZ.U.G.
A NON PROFIT ORGANIZATION

FRONT COVER

JASON OAKLEY HAS BEEN BUSY ON THE FRONT COVER ONCE MORE. AGAIN IT RELATES TO AN ARTICLE INSIDE. THANKS JASON.

DISCLAIMER: EVERY EFFORT IS MADE TO INSURE THE ACCURACY OF INFORMATION CONTAINED WITHIN BE IT GENERAL, TECHNICAL, PROGRAMMING, ETC. NO RESPONSIBILITY CAN BE ACCEPTED BY HUNTER VALLEY VZ USERS' GROUP OR AUTHOR AS A RESULT OF APPLYING SUCH INFORMATION IN PRACTICE.

APOLOGIES:

I STARTED JOURNAL EARLY JULY WHEN THINGS GOT WORSE MEDICALLY. MY CONTINUING MEDICAL PROBELM IS THE RESULT OF THE CAR ACCIDENT OVER 4 YEARS AGO WHICH AFFECTED MY SPINE AND EVERY JOINT IN MY BODY. THE CHRONIC PAINS ARE USUALLY AT A BEARABLE LEVEL, BUT THE WINTER JUST PASSED HAS NOT HELPED.

TO COMPOUND THE PROBLEM, PAIN KILLERS GIVE NO RELIEF AT ALL. THE ONLY THING THAT REDUCES THE CHRONIC PAINS IS LYING DOWN SEVERAL TIMES A DAY WHICH TAKES A BIG CHUNK OUT OF THE DAY. IN THE EVENT OF THE JOURNAL CEASING PRODUCTION OUTSTANDING SUBSCRIPTIONS WOULD BE REFUNDED AS THE CLUB'S FUNDS ARE QUITE SOUND.

IT IS MY INTENTION TO PRUDUCE THE JOURNAL FOR SOME TIME TO COME AS THERE IS STILL A LOT OF LIFE LEFT IN THE HUMBLE VZ. IT GIVES ME A GREAT DEAL OF PLEASURE IN PRUDUCING THE JOURNAL AND BEING PART OF A GROUP OF PEOPLE AUSTRALIA WIDE.

GET WELL SOON NEVILLE:

ON BEHALF OF THE HUNTER VALLEY VZ USER'S GROUP AND MYSELF WE WISH NEVILLE HUGHES A SPEEDY RECOVERY FROM 3 VERY BAD HEART ATTACKS. KEEP UP THE GOOD FIGHT NEVILLE AND WE LOOK FORWARD TO YOUR NEXT PROJECT.

TECHNICAL DATA SHEETS:

I WOULD LIKE YOUR OPINION ON THE DATA SHEETS AS I INTEND THEM TO BECOME A REGULAR ITEM IN THE JOURNAL. IF YOU HAVE A CONTRIBUTION OR IDEAS FOR THEM, THEN PLEASE LET ME KNOW, ED.

FOR SALE - IBM MOTHERBOARD:

A COUPLE WEEKS AGO I INSTALLED AN UP-GRADEABLE 386DX MOTHERBOARD WITH 4M RAM IN MY IBM COMPATIBLE PC. NEEDLESS TO SAY THE OLD MOTHERBOARD IS NO LONGER REQUIRED AND I OFFER IT FOR SALE.

1 OFF BABY AT 286 MOTHERBOARD - 6/12 MHZ
1 MEG RAM AND MATHS CO-PROCESSOR ON BOARD
RAM EXPANDABLE TO 4M USING 4 X 1M SIPPS
1 OFF INSTRUCTION MANUAL

PRICE: $150.00 ONO- INCLUDES POST & PACKING - CONTACT:
JOSEPH P LEON - PHONE (049) 692-399 - 33 TIGHES TCE TIGHES HILL 2297

NEXT ISSUE:

IT IS MY INTENTION TO PRODUCE ANOTHER ISSUE AND HAVE IT POSTED OUT BEFORE CHRISTMAS 1992 AND HOPEFULLY KEEP TO A BI-MONTHLY SCHEDULE. BELOW ARE SOME OF THE PROPOSED MATERIAL FOR NEXT ISSUE.

1) LISTING OF PUBLIC DOMAIN PROGRAMS.
2) LISTING OF PUBLIC DOMAIN SOURCE CODE.
3) ARTICLE ON EDITOR/ASSEMBLER INCOMPATIBILITY.
4) TECHNICAL DATA SHEET # 4, (5-6?).
5) PLUS CONTINUATION OR CONCLUSION OF PARTLY PUBLISHED MATERIAL.

VZ BUS MOUSE PROJECT PART 1
BY LESLIE MILBURN

THIS PROJECT IS AN ALTERNATIVE TO THE MOUSE PROJECT PRESENTED IN THE LAST JOURNAL AND ALLOWS A MOUSE TO BE CONNECTED WITHOUT MODIFICATIONS TO THE VZ. ALL MICE WORK ROUGHLY THE SAME WAY SO I SHALL NOT REPEAT HERE WHAT WAS DESCRIBED IN GARY BULLEY'S ARTICLE. THERE ARE TWO TYPES OF MOUSE:-

(1) A SERIAL MOUSE - THIS IS THE MOST COMMON TYPE OF MOUSE AND HAS AN RS232C INTERFACE. THIS CAN HAVE EITHER A DB9 OR DB25 CONNECTOR. THIS IS THE TYPE OF MOUSE WHICH CAN BE MODIFIED FOR THE VZ USING GARY'S METHOD. AN ALTERNATIVE SOLUTION IS TO BUILD AN RS232 PORT FOR THE VZ AND PLUG IT IN DIRECTLY.

(2) A BUS MOUSE - THIS TYPE OF MOUSE IS USED BY THE AMIGA, AMSTRAD, ATARI AND SOME IBM CLONES SUCH AS THE COMMODORE PC COLT, PC10 & PC20 AND NOW THE VZ. THE PLUG ON THIS TYPE OF MOUSE IS A DB9 PLUG. FOR THOSE WHO NEED HELP FINDING A SUITABLE MOUSE THERE IS ONE AVAILABLE FOR $50-00 (APPROX) FROM GRACE BROS. IT IS A QTRONIX MOUSE FOR THE AMIGA 500/1000/2000, SERIAL NUMBER X30B (BUS VERSION). HOWEVER, SHOP AROUND AS I'M SURE THAT YOU CAN BUY THEM CHEAPER.

FOR EACH OF THESE MICE, A TWO BUTTON AND THREE BUTTON VERSION EXISTS.

THE HARDWARE:-

TO CONNECT A BUS MOUSE TO THE VZ, THE 8 BIT INPUT PORT AS DESCRIBED IN JOURNAL #30 IS REQUIRED. THE PIN DEFINITIONS ARE AS FOLLOWS:-

PIN - DESCRIPTION	DATA LINE
1 - VERTICAL MOTION	- D0
2 - HORIZONTAL MOTION	- D1
3 VERTICAL MOTION 90 DEG. OUT OF PHASE OF 1.	- D2
4 - HORIZONTAL MOTION 90 DEG. OUT OF PHASE OF 2.	- D3
5 - MOUSE BUTTON 3 (OPTIONAL)	- D4
6 - MOUSE BUTTON 1	- D5
7 - +5 VOLTS	- N/A
8 - GROUND	- N/A
9 - MOUSE BUTTON 2	- D6

AS CAN BE SEEN ABOVE, SOME MODIFICATIONS TO THE INPUT PORT MUST BE MADE. THE CHANGES I MADE WERE TO USE A DB9 SOCKET AND CONNECT THE PINS AS SHOWN ABOVE. D7 MUST BE CONNECTED TO GROUND, THIS IS REQUIRED SO THAT THE MOUSE DRIVER CAN DETECT THE PRESENCE OF THE MOUSE PORT.

<u>WARNING:</u> WHEN CONNECTING THE 8 BIT INPUT PORT TO THE PRINTER INTERFACE, CHECK WHICH VERSION OF THE INTERFACE YOU HAVE. THIS IS BECAUSE SOME INTERFACES USE A4 TO TRIGGER THE 74LS138 NOT A0.

THIS MEANS THAT ALTHOUGH THE PRINTER IS ACCESSED WHEN A4 IS LOW (I.E. PORTS 0 - 15), THE INPUT PORT IS ACCESSED WHEN A4 IS HIGH (I.E. PORTS 16 - 31). THIS CAUSES ADDRESS CONFLICT WITH THE DISK SYSTEM. ALSO NOTE THAT THIS VERSION OF THE INTERFACE IS NOT IN THE VZ TECHNICAL MANUAL. BE CAREFUL!!

THE MOUSE DRIVER:-

THE VZ MOUSE DRIVER IS AN INTERRUPT DRIVEN MACHINE CODE PROGRAM WHICH INTERPRETS THE RAW DATA SENT TO THE VZ BY THE MOUSE. THIS RAW DATA INDICATES MOUSE MOVEMENT AND BUTTON PRESSES.

THE MAIN AIM OF THIS PROJECT IS TO PROVIDE MOUSE FACILITIES EQUIVALENT TO THOSE ON AN IBM PC. THE LATEST MICROSOFT MOUSE DRIVER (VERSION 8.0) PROVIDES 53 FUNCTION CALLS THAT LET PROGRAMS PERFORM SPECIFIC TASKS, SUCH AS CHECKING A MOUSE BUTTON. THE VZ MOUSE DRIVER PROVIDES ALMOST THE WHOLE SUITE OF MOUSE FUNCTIONS.

THE MOUSE DRIVER CAN BE BROKEN INTO THREE DISTINCT PARTS:-

(1) THE INTERRUPT ROUTINE,
(2) THE MOUSE FUNCTION LIBRARY,
(3) THE BASIC MOUSE FUNCTION.

NOTE:

(I) VERSION 1.03 OF THE VZ MOUSE DRIVER ONLY SUPPORTS TEXT MODE(0).
(II) THIS VERSION OF THE MOUSE DRIVER SUPPORTS A MOUSE CURSOR WHICH IS INDEPENDENT OF THE KEYBOARD CURSOR.

THE INTERRUPT ROUTINE:-

EVERYTIME AN INTERRUPT OCCURS THIS ROUTINE MUST BE CALLED. IT LOOKS AFTER THE POLLING OF THE MOUSE AND MAINTENANCE OF ALL MOUSE "VARIABLES". THE MORE TIMES THE MOUSE IS POLLED, THE MORE SENSITIVE THE MOUSE CURSOR MOVEMENT. THE NUMBER OF TIMES THE MOUSE IS POLLED CAN BE SET BY A PROGRAM.

THE MOUSE FUNCTION LIBRARY:-

AS MENTIONED ABOVE, THERE ARE 53 IBM MOUSE FUNCTIONS. NOT ALL FUNCTIONS ARE PROVIDED IN V1.03 OF THIS DRIVER. THESE FUNCTIONS ARE DOCUMENTED IN TWO READILY AVAILABLE BOOKS:-

(1) THE MICROSOFT MOUSE PROGRAMMERS REFERENCE - $65-00 FROM DYMOCKS
(2) THE DOS PROGRAMMERS REFERENCE - $60-00 (APPROX) ALSO FROM DYMOCKS.

IN THE NEXT JOURNAL I SHALL DESCRIBE EACH OF THE AVAILABLE FUNCTIONS IN DETAIL.

THE BASIC MOUSE FUNCTION:-

TO ALLOW THE BASIC PROGRAMMER EASY ACCESS TO THE MOUSE FUNCTION LIBRARY, A NEW BASIC COMMAND HAS BEEN PROVIDED. THE SYNTAX OF THIS COMMAND IS:-

MOUSE(M1%, M2%, M3%, M4%).

M1% IS THE MOUSE FUNCTION NUMBER.
M2% - M4% ARE PARAMETERS TO THE MOUSE FUNCTIONS.

NOTE: NOT ALL PARAMETERS ARE USED BY ALL MOUSE FUNCTIONS BUT THEY MUST STILL BE PROVIDED IN THE CALL.

THE NEW COMMAND MAY BE CALLED FROM WITHIN A BASIC PROGRAM OR IN IMMEDIATE MODE. ALTHOUGH YOU MAY USE INTEGER CONSTANTS AND VARIABLES TO CALL MOUSE FUNCTIONS, RETURN VALUES ARE PASSED BACK TO THE PROGRAM. THESE RETURN VALUES WILL ALWAYS BE STORED IN THE VARIABLES M1%, M2%, M3% AND M4%, AND AS SUCH PROGRAMS SHOULD AVOID STORING DATA IN THESE VARIABLES.

TO BE CONTINUED ....

```
;****************************
;THIS PROGRAM IS A SAMPLE
;USER FUNCTION TO BE USED
;IN CONJUNCTION WITH KSCAN
;****************************
BOX  EQU  28866
SCAN EQU  002BH
PWON EQU  4008H
PWOF EQU  400BH
SRCH EQU  402CH
LOAD EQU  4041H
FTYP EQU  "T"
NEW  EQU  1B4DH
RUN  EQU  36E9H
EXIT EQU  1A19H
WLTC EQU  6800H
RLTC EQU  30779
;
;****************************
;USER: APPLICATION PROVIDED
;        USER FUNCTION.
;
USER PUSH AF
     CP   2
     JR   NZ,USE4
     CALL SAVE
     CALL DBOX
     LD   HL,MES1
USE1 CALL MSG
USE2 CALL SCAN
     CP   "N"
     JR   Z,USE3
     CP   "Y"
     JR   NZ,USE2
;LOAD THE DISK FILENAME
     CALL LNAM
;CHECK IF THE FILE EXISTS
     CALL FIND
     LD   HL,MES2
     CP   2
     JR   NZ,USE1
;CLEAR THE PROGRAM AREA
     CALL NEW
     CALL PWON
     CALL LOAD
     PUSH AF
     CALL PWOF
     POP  AF
     LD   HL,MES3
     CP   0
     JR   NZ,USE1
     LD   DE,(30884)
     JP   RUN
USE3 CALL REST
     POP  AF
     XOR  A
     RET
USE4 POP  AF
     RET
;****************************
;DBOX: DRAW THE POPUP MENU.
;
DBOX LD   HL,BOX
     LD   (HL),142
     INC  HL
     LD   (HL),140
     LD   DE,BOX+2
     LD   BC,26
     LDIR
     LD   (HL),141
     LD   HL,BOX+64
     LD   (HL),139
     INC  HL
     LD   (HL),131
     LD   DE,BOX+66
     LD   BC,26
     LDIR
     LD   (HL),135
     RET
;****************************
;MSG: DISPLAY USER MESSAGE.
;
MSG  LD   DE,BOX+32
     LD   BC,28
     LDIR
     RET
MES1 EQU  $
:3
     DEFB 138
*GO TO THE MAIN MENU (Y/N)?*
     DEFB 133
MES2 EQU  $
:3
     DEFB 138
*ERROR: CHANGED DISK (Y/N)?*
     DEFB 133
MES3 EQU  $
:3
     DEFB 138
* DISK ERROR: RETRY (Y/N)? *
     DEFB 133
;****************************
;SAVE: SAVE THE SCREEN INFO
;      WHERE THE POP UP IS
;      TO BE DISPLAYED.
;
SAVE LD   HL,BOX
     LD   DE,BOX1
     LD   BC,96
     LDIR
     LD   A,32
     LD   (WLTC),A
     RET
BOX1 DEFS 100
;****************************
;REST: THIS RESTORES THE
;      DISPLAY BACK TO ITS
;      ORIGINAL STATE.
;
REST LD   HL,BOX1
     LD   DE,BOX
     LD   BC,96
     LDIR
     LD   A,(RLTC)
```

```
        LD   (WLTC),A
        RET
;****************************
;LNAM: LOAD THE NAME OF THE
;      PROGRAM INTO THE DISK
;      FILENAME BUFFER.
;
LNAM LD   HL,FNAM
     PUSH IY
     POP  DE
     INC  DE
     LD   BC,8
     LDIR
     LD   A,FTYP
     LD   (DE),A
     RET
FNAM EQU  $
:1
*MAINMENU*
;****************************
;FIND: CHECK THAT THE DISK
;      CONTAINS THE PROGRAM
;      TO LOAD.
;
FIND CALL PWON
     CALL SRCH
     PUSH AF
     CALL PWOF
     POP  AF
     RET
```

EDITOR'S COMMENTS:

THE SOURCE CODE FOR KSCAN2.1 CONSISTS OF 5 FILES, INIT0, BASIC, SYSTEM, PLOT AND MATRIX. BELOW YOU WILL FIND DETAILS OF EACH FILE TOGETHER WITH NUMBER OF LINES OF EACH FILE AND NUMBER OF PAGES PER FILE WHEN PRINTED IN TWO COLUMNS PER PAGE (SEE PAGE 6 FOR EXAMPLE).

IF I PRINTED 3/4 PAGES PER ISSUE IT WOULD TAKE 4/5 ISSUES TO COMPLETE WHICH IS TOTALLY UN-ACCEPTABLE.

ONLY ONE EDITOR ASSEMBLER IS CAPABLE OF ASSEMBLING THE SOURCE CODE, LESLIE MILBURN'S DISKOPS6 (64K VERSION - PUBLIC DOMAIN) MUST BE USED. BESIDES NEEDING THE EXTRA MEMORY DISKOPS6 ALSO HAS ADDITIONAL COMMANDS, ONE OF WHICH IS USED TO 'LINK' THE SOURCE CODE TOGETHER.

```
INIT0    13 05 A813 DEEF 36DC - LINK BASIC  - 694 LINES -  5.60 PAGES
BASIC    1A 05 A813 C831 201E - LINK SYSTEM - 451 LINES -  3.63 PAGES
SYSTEM   0B 0C A813 B7E7 0FD4 - LINK PLOT   - 180 LINES -  1.45 PAGES
PLOT     0D 0D A813 AA8A 0277 - LINK MATRIX -  21 LINES -  0.16 PAGES
MATRIX   0E 03 A813 D04D 283A -             - 599 LINES -  4.83 PAGES
                                            -------------------------
                                            1945 LINES - 15.68 PAGES

USER     02 00 A813 B34B 0B38 - PRINTED THIS ISSUE.
KSCAN2.1 01 00 7AE9 8267 077E - KSCAN2.1 OBJECT CODE (M/C)
```

FOR ANY INFORMATION ON KSCAN2.1, ITS SOURCE CODE OR DISKOPS6 CONTACT LESLIE MILBURN. ADDRESS AND PHONE NUMBER ON PAGE 12.

OPERATING INSTRUCTIONS FOR EXTENDED DOS FACILITIES V12.1 BY LESLIE MILBURN

EXT DOS V12.1 COMMANDS

FORMAT

THIS IS THE SAME AS DOS INIT. IT INITIALISES A DISK FOR USE. WHEN FINISHED IT PROMPTS USER FOR A DISK LABEL AND THE CURRENT DATE.

LABEL - THIS ALLOWS THE USER TO PUT A LABEL ON A FORMATTED DISK.

VOL - THIS ALLOWS THE USER TO READ BACK THE DISK LABEL.

NOTE: THE DISK LABEL USES A S:FILETYPE (SYSTEM FILE) AND OCCUPIES ONE SECTOR ON THE FLOPPY DISKETTE. IF YOU WISH TO PLACE THE DISK LABEL ON PART OF THE DISK WHICH IS NOT USED THEN I SUGGEST THAT THE SECOND HALF OF TRACK 0 SECTOR 15 BE USED. THIS IS CURRENTLY UNUSED.

DIS? - (?) = FILETYPE (OPTIONAL)

THIS DISPLAYS ALL FILES ON THE FLOPPY DISKETTE BY ?:FILETYPE. THE (?) IS OPTIONAL BUT IF NOT SPECIFIED THEN ALL FILES WILL BE DISPLAYED. IF T IS SPECIFIED, (IE. DIST) THEN ONLY T:FILETYPES WILL BE DISPLAYED. THE OUTPUT FORMAT IS AS FOLLOWS:

```
FILETYPE:FILENAME -     START     - MEMORY ADDRESS
                    TRACK SECTOR -  START END
      T:FILENAME      01  00         7AE9 8673
```

AFTER EACH SECTOR OF THE DISK DIRECTORY HAS BEEN DISPLAYED THE OUTPUT WILL BE PAUSED. PRESS <u>SPACE</u> TO CONTINUE LISTING OR <u>BREAK</u> TO EXIT.

CHA"FILENAME",? - (?) = FILETYPE

THIS FACILITY ALLOWS YOU TO CHANGE THE FILETYPE OF A FILE ON DISK. THE FILENAME AND FILETYPE ARE REQUIRED.

NOTE: AFTER CHANGING THE FILETYPE THE PROGRAM OR DATA MAY NO LONGER RUN OR LOAD. CHA CAN BE USED TO PROTECT FILES AGAINST ACCIDENTAL ERASURE OR TO CONVERT SOURCE CODE FILES FOR USE WITH OTHER EDITOR ASSEMBLERS.

REL"FILENAME",XXXX

THIS FACILITY ALLOWS YOU TO RELOCATE A FILE. THE FILENAME AND NEW START ADDRESS IS REQUIRED. A NEW END ADDRESS IS AUTOMATICALLY CALCULATED AND BOTH THE NEW START AND END ADDRESSES ARE PLACED INTO THE DISK DIRECTORY. USE HEX NUMBERS ONLY.

NOTE: A BASIC PROGRAM CAN BE RELOCATED TO A NEW START ADDRESS. HOWEVER, SOMETIMES THE RUN COMMAND WILL NOT WORK. HINT: TRY TYPING THE FIRST LINE IN IMMEDIATE MODE THEN GOTO 20 (FOR EXAMPLE).

EXA,XXXX,YYYY

THIS FACILITY ALLOWS YOU TO EXAMINE ANY TRACK (XXXX) AND SECTOR (YYYY) ON THE FLOPPY DISKETTE. USE ONLY HEX NUMBERS. THE SECTOR WILL THEN BE DISPLAYED. PRESS <u>SPACE</u> TO CONTINUE SEEING THE FOLLOWING SECTORS ON THE DISKETTE. TO EXIT USE <u>BREAK</u>.

```
;****************************
;EXTENDED DOS COMMANDS V12.1
;
;CODE IS NOT RELOCATABLE.
;****************************
PCHR EQU  033AH
SYER EQU  1997H
GCHR EQU  1D78H
PMSG EQU  2B75H
PWON EQU  4008H
PWOF EQU  400BH
DERR EQU  400EH
SRCH EQU  402CH
SECT EQU  4035H
CSI  EQU  401DH
WRTE EQU  4032H
INBF EQU  79E8H
INPT EQU  0567H
FSAV EQU  4044H
COMA EQU  44
HEX  EQU  4020H
CLS  EQU  01C9H
WKEY EQU  49H
BAS  EQU  1A19H
NEW  EQU  1B4DH
RST  EQU  7804H
ORG  EQU  7AE9H
SBAS EQU  78A4H
TOM  EQU  78B1H
HTAB EQU  78A6H
SSPC EQU  78A0H
;
;**************************
;BRUN: THIS FUNCTION IS
;       CALLED WHEN RUN FROM
;       DISK.
        CALL CLS
BRUN    LD   SP,7700H
        LD   HL,(RST)
        LD   (VCTR),HL
        LD   HL,CMD
        LD   (RST),HL
        DEC  HL
        LD   (TOM),HL
        LD   DE,50
        OR   A
        SBC  HL,DE
        LD   (SSPC),HL
        LD   HL,ORG
        LD   (SBAS),HL
        CALL NEW
        JP   BAS
;**************************
;CMD: THIS FUNCTION CHECKS
;       AND PROCESSES THE NEW
;       COMMANDS.
CMD     EXX
        LD   HL,1D5BH
        POP  DE
        OR   A
        SBC  HL,DE
        PUSH DE
        EXX
        JP   NZ,GCHR
        PUSH HL
        CALL GCHR
        JR   NZ,CMD1
POP     POP  HL
        LD   DE,(VCTR)
        PUSH DE
        RET
CMD1    CP   "V"
        JR   NZ,CMD2
        CALL VOL
        JR   C,POP
        JR   CMDX
CMD2    CP   "L"
        JR   NZ,CMD3
        CALL LABL
        JR   C,POP
        JR   CMDX
CMD3    CP   "E"
        JR   NZ,CMD4
        CALL EXA
        JR   C,POP
        JR   CMDX
CMD4    CP   "D"
        JR   NZ,CMD5
        CALL DIS
        JR   C,POP
        JR   CMDX
CMD5    CP   "C"
        JR   NZ,CMD6
        CALL CHA
        JR   C,POP
        JR   CMDX
CMD6    CP   "R"
        JR   NZ,POP
        CALL REL
        JR   C,POP
CMDX    POP  DE
        POP  DE
        LD   DE,1D1EH
        PUSH DE
        RET
VCTR    DEFW 0
FNAM    EQU  $
:1
*DISKINFO*
;****************************
;VOL: THIS DISPLAYS THE DISK
;       LABEL.
;
VOL     CALL GCHR
        CP   "O"
        SCF
        RET  NZ
        CALL GCHR
        CP   "L"
        SCF
        RET  NZ
        CALL GCHR
        JP   NZ,SYER
;SAVE THE BUFFER POINTER
```

```
        PUSH HL
;LOAD THE FILENAME
        CALL LNAM
        DI
        CALL PWON
        CALL SRCH
        CP   2
        LD   A,13
        JP   NZ,DERR
;UPON EXIT FROM SEARCH DE
;POINTS TO START TRACK IN
;DIR ENTRY.
        LD   A,(DE)
        LD   (IY+18),A
        INC  DE
        LD   A,(DE)
        LD   (IY+17),A
        CALL SECT
        CALL PWOF
        EI
;CLEAR TWO SCREEN LINES.
        LD   HL,(7820H)
        PUSH HL
        LD   B,63
        LD   A,32
VOL1    CALL PCHR
        DJNZ VOL1
        POP  HL
        LD   (7820H),HL
        LD   A,0
        LD   (HTAB),A
;GET POINTER TO DISK LABEL
        LD   H,(IY+50)
        LD   L,(IY+49)
;SAVE IT
        PUSH HL
;PLACE A NULL BYTE AFTER
;EACH STRING. THIS MEANS
;THAT DISK LABELS CREATED BY
;QUICKWRITE CAN BE DISPLAYED
        PUSH HL
        LD   DE,62
        ADD  HL,DE
        LD   (HL),0
        ADD  HL,DE
        INC  HL
        INC  HL
        ADD  HL,DE
        LD   (HL),0
;DISPLAY DISK LABEL.
        POP  HL
        CALL PMSG
        LD   A,13
        CALL PCHR
        POP  HL
        LD   DE,63
        ADD  HL,DE
        CALL PMSG
        LD   A,13
        CALL PCHR
        POP  HL
        OR   A
        RET
;****************************
;LNAM: THIS FUNCTION LOADS
;      THE FILENAME INTO THE
;      DISK NAME BUFFER.
LNAM    LD   HL,FNAM
        PUSH IY
        POP  DE
        INC  DE
        LD   BC,8
        LDIR
        LD   (IY+9),"S"
        RET
;****************************
;LABL: THIS FUNCTION PLACES
;      A LABEL ON A DISK.
LABL    CALL GCHR
        CP   "A"
        SCF
        RET  NZ
        CALL GCHR
        CP   "B"
        SCF
        RET  NZ
        CALL GCHR
        CP   "E"
        SCF
        RET  NZ
        CALL GCHR
        CP   "L"
        SCF
        RET  NZ
        CALL GCHR
        JP   NZ,SYER
;SAVE BUFFER POINTER.
        PUSH HL
;SAVE INPUT BUFFER CONTENTS.
        LD   HL,INBF
        LD   DE,LBF2
        LD   BC,64
        LDIR
        LD   HL,MES1
        CALL PMSG
LAB1    CALL INPT
        JP   C,LABX
        LD   A,(INBF)
        CP   32
        JR   Z,LAB1
;COPY INPUT TO INTERNAL
;BUFFER.
        LD   DE,LBF1
        LD   HL,INBF
        LD   BC,62
        LDIR
        LD   HL,MES2
        CALL PMSG
LAB2    CALL INPT
        JP   C,LABX
        LD   A,(INBF)
        CP   32
        JR   Z,LAB2
;COPY INPUT TO INTERNAL
```

```
;BUFFER.
        LD    DE,LBF1+63
        LD    HL,INBF
        LD    BC,62
        LDIR
;PLACE LABEL NAME INTO
;DISK BUFFER.
        CALL LNAM
        DI
        CALL PWON
;CHECK IF THE FILE ALREADY
;EXISTS.
        CALL SRCH
        CP    2
        JP    Z,DERR
;SAVE THE CURRENT START AND
;END OF BASIC POINTERS.
        LD    HL,(78A4H)
        PUSH HL
        LD    HL,(78F9H)
        PUSH HL
;SET THE START AND END
;ADDRESS OF LABEL DATA.
        LD    HL,LBF1
        LD    (78A4H),HL
        LD    HL,LBF1+126
        LD    (78F9H),HL
        CALL FSAV
        CALL PWOF
        EI
;RESTORE ORIGINAL START AND
;END OF BASIC POINTERS.
        POP   HL
        LD    (78F9H),HL
        POP   HL
        LD    (78A4H),HL
LABX    LD    HL,LBF2
;RESTORE INPUT BUFFER CONTS.
        LD    DE,INBF
        LD    BC,64
        LDIR
        POP   HL
        OR    A
        RET
MES1    EQU   $
:1
*ENTER DISK LABEL:-*
        DEFB 13
        DEFB 0
MES2    EQU   $
*ENTER TODAY'S DATE:-*
        DEFB 13
        DEFB 0
;INTERNAL BUFFER FOR STORING
;THE DISK LABEL.
LBF1    DEFS 126
LBF2    DEFS 64
;****************************
;EXA: THIS FUNCTION ALLOWS
;       ANY SECTOR TO BE
;       VIEWED.
EXA     CALL GCHR
        CP    "X"
        SCF
        RET   NZ
        CALL GCHR
        CP    "A"
        SCF
        RET   NZ
        CALL GCHR
        CP    COMA
        SCF
        RET   NZ
;INCREMENT POINTER PAST THE
;COMMA.
        INC   HL
        CALL HEX
        RET   C
        PUSH HL
        LD    HL,39
        RST   18H
        POP   HL
        RET   C
        LD    (IY+18),E
;CHECK NEXT CHAR.
        LD    A,(HL)
        CP    COMA
        SCF
        RET   NZ
        INC   HL
        CALL HEX
        RET   C
        PUSH HL
        LD    HL,16
        RST   18H
        POP   HL
        RET   C
        LD    (IY+17),E
        DEC   HL
        CALL GCHR
        JP    NZ,SYER
;SAVE BUFFER POINTER
        PUSH HL
        CALL CLS
EXA1    DI
        CALL PWON
        CALL SECT
        CALL PWOF
        EI
;DISPLAY THE SECTOR.
        CALL DSCT
        LD    A,(IY+17)
        INC   A
        CP    16
        JR    Z,EXA4
EXA2    LD    (IY+17),A
EXA3    CALL WKEY
        CP    1
        JR    Z,EXAX
        CP    32
        JR    Z,EXA1
        JR    EXA3
EXA4    LD    A,(IY+18)
        INC   A
```

CONTINUED NEXT ISSUE . . .

DISKOPS1 & 2 (DISK OPERATIONS) 40/12

INSTRUCTIONS BY LESLIE MILBURN

THIS IS A MACHINE CODE UTILITY WHICH ALLOWS MACHINE CODE PROGRAMMERS USING THE DICK SMITH EDITOR ASSEMBLER TO SAVE AND LOAD THEIR SOURCE CODE TO AND FROM DISK. (NOTE TO SAVE OBJECT CODE A TAPE MUST STILL BE USED) AS THE PROGRAM IS NOT RELOCATABLE TWO VERSIONS ARE SUPPLIED ON DISK, THESE ARE:-

DISKOPS1 - EXPANDED VZ200
DISKOPS2 - EXPANDED VZ300

GETTING THE PROGRAMS OPERATIONAL:-

STEP 1: POKE 30897,200:POKE 30898,242
STEP 2: BRUN"EDITOR ASSEMBLER"
OR: CRUN"EDITOR ASSEMBLER"
STEP 3: ONCE THE EDITOR ASSEMBLER IS OPERATIONAL ENTER THE FOLLOWING LINE:-
JP 4005H - ASSEMBLE AND RUN. THIS WILL REBOOT THE DOS.
STEP 4: BRUN"DISKOPS 1 OR 2" DEPENDING ON THE MEMORY SIZE OF YOUR VZ.

THE ADDITIONAL COMMANDS ARE:-

TU:FILENAME - LOAD SOURCE FILE FROM DISK.
TR:FILENAME - SAVE SOURCE FILE ON DISK.
TD: - THIS IS FOR USE WHEN THE SOURCE FILE ISN'T COMPLETE. THE SOURCE CODE IS SAVED WITH THE FILENAME OF SOURCE. IF THE FILE ALREADY EXISTS THEN THE NEW FILE REPLACES THE FILE ON DISK.

WARNING:-

THE TD: ROUTINE INTERFERES WITH THE DISK TRACK MAP SO USE THIS WITH CAUTION. I RECOMMEND THAT A SEPARATE DISK BE USED AS IT IS POSSIBLE FOR DISK FILES TO BE DAMAGED.

NOTE:-

ALL SOURCE CODE FILES WHEN SAVED ARE OF FILETYPE 'W' AND ONLY THE FIRST 8 CHARACTERS OF THE FILENAME ARE RECOGNISED, IE: W:FILENAME

ERROR MESSAGES:-

IF THE ORIGIN IS NOT SET TO THE FIRST FREE BYTE AFTER THE SOURCE CODE USING THE 'O' <<RETURN>> COMMAND THEN YOU WILL GET A *** WRONG ORIGIN FOR RUN *** ERROR.

INFORMATION AND SOURCE CODE:-

WRITE TO LESLIE MILBURN:- FOR A SOURCE CODE LISTING OR IF YOU HAVE ANY OTHER PROBLEMS.

LESLIE MILBURN 11 SWORDFISH AVENUE RABY 2566 PHONE (02) 603-5981

A SOURCE CODE LISTING IS AVAILABLE FOR $5-00.

LISTING 5:

```
0010 '**************************
0020 '***  1536 VZ SCREENS   ***
0030 '***    VERSION 1.3     ***
0040 '***  R.B.K.  18/5/86   ***
0050 '**************************
0088 '
0089 '***FIND TOP OF MEMORY AND PROTECT 20 BYTES BY LOWERING TOM.
0090 IF PEEK (30791)=77 THEN GOTO 500       :'7847H AS LOAD CHECK.
0100 M1=PEEK(30898):L1=PEEK(30897)          :'TOM POINTERS.
0110 TM=M1*256+L1-21                        :'PROTECT 20 BYTES.
0120 MS=INT(TM/256):LS=TM-MS*256            :'NEW TOM POINTERS.
0130 POKE 30898,MS:POKE 30897,LS            :'SET NEW TOM.
0140 CLEAR 50                               :'RESET ALL POINTERS.
0150 'THIS ACTION DESTROYS THE VLT-ALL VARIABLES MUST BE RENEWED.
0160 M1=PEEK(30898):L1=PEEK(30897)          :'NEW TOM POINTERS.
0170 ST=M1*256+L1+1                         :'NEXT ADDR IN RESERVED
0180 MS=INT(ST/256):LS=ST-256*MS            :'START OF ROUTINE.
0190 IF ST>32767 THEN ST=ST-65536           :'CONVERT TO SIGNED INT
0198 '
0199 '***LOAD MACHINE CODE INTO PROTECTED TOM AND SET USR POINTER
0200 CLS:PRINT@230,"LOADING MACHINE CODE"
0210 CS%=0                                  :'CHECK SUM COUNTER.
0220 FOR ID=ST TO ST+15                     :'LOAD 16 BYTES TO TOM.
0230    READ VL%:POKE ID,VL%:CS%=CS%+VL%    :'LOAD M/L INTO PROTECT
0240 NEXT ID
0250 IF CS%<>1483 THEN GOTO 1000            :'WRONG DATA.
0260 POKE 30863,MS:POKE 30862,LS            :'SET USR POINTERS.
0270 POKE 30791,77                          :'SET LOAD CHECK.
0399 '
0400 '***Z-80 BLOCK MOVE ROUTINE.
0410 DATA 058,033,121   :'LD A,(7921H) PICK UP VALUE FROM USR().
0420 DATA 033,000,112   :'LD HL,7000H  (#28672D START VIDEO RAM)
0430 DATA 017,001,112   :'LD DE,7001H  (#28673D NEXT)
0440 DATA 001,255,007   :'LD BC,07FFH  (#2047D  SIZE OF VIDEO RAM)
0450 DATA 119           :'LD (HL),A
0460 DATA 237,176       :'LDIR         (BLOCK LOAD COMMAND)
0470 DATA 201           :'RET
0498 '
0499 '***INITIALIZE DELAYS - CONTROL SPEED OF EXECUTION BY D.
0500 T%=0                                   :'TONE IS REST (0-31)
0510 D%=4                                   :'DELAY IS SHORT(1-9)
0520 P%=30744                               :'SET UP SCREEN MODE.
0594 '
0595 '***THIS LOOP DOES A LOT OF WORK.
0596 '***IT LOADS ALL VALUES INTO THE A-REG FOR VIEWING.
0597 '***IT SWAPS BETWEEN HI- AND LO-RES SCREEN MODES.
0598 '***IT SWITCHES BETWEEN NORMAL AND INVERSE SCREEN TYPES.
0599 '***IT SWITCHES BETWEEN GREEN AND ORANGE BACKGROUNDS.
0600 FOR A%=0 TO 255          :'ALL VALUES PUT INTO A=REG.
0610    MODE(0):POKE P%,0
0620    CLS:PRINT@234,"VALUE = ";A%:SOUND T%,D%
0630    X=USR(A%)
0640    COLOR,0:SOUND T%,D%
0650    COLOR,1:SOUND T%,D%
0660    POKE P%,1
0670    X=USR(A%)
0680    COLOR,0:SOUND T%,D%
0690    COLOR,1:SOUND T%,D%
0699 '
```

```
0700 '  HI-RES SCREENS.
0710    MODE(1)
0720    X=USR(A%)
0730    COLOR,0:SOUND T%,D%
0740    COLOR,1:SOUND T%,D%
0750 NEXT A%
0800 GOTO 600
0998 '
0999 '***WRONG DATA MESSAGE.
1000 CLS:PRINT@230,"ERROR IN DATA ";CS%
1010 END
```

FUNCTIONS IN BASIC

IT MAY BE A SURPRISE TO MOST BASIC PROGRAMMERS BUT THE FUNCTION (FN) COMMAND, ALONG WITH THE SUBROUTINE (GOSUB) COMMAND, ARE PROBABLY THE MOST USEFUL BASIC COMMANDS. THEY ARE CONCISE AND CLARIFY BASIC CODING CONSIDERABLY. UNFORTUNATELY ONLY GOSUB IS SUPPORTED ON THE VZ.

I HAVE HAD A NUMBER OF QUERIES FROM USERS ON HOW TO USE THE FN STATEMENT - PARTICULARLY WHEN TRYING TO CONVERT PROGRAMS WRITTEN FOR OTHER MACHINES. IN THIS BRIEF NOTE, I WILL TRY TO EXPLAIN HOW TO GET AROUND IT.

IN THE LEVEL II BASIC THAT IS IN THE VZ'S ROM, TWO TYPES OF FUNCTION ARE SUPPORTED -

1. LIBRARY (OR SYSTEM) FUNCTIONS, AND
2. USER-DEFINED FUNCTIONS.

FUNCTIONS CAN BE USED TO MANIPULATE NUMERIC OR STRING DATA TYPES. THE VZ SUPPORTS A NUMBER OF INTRINSIC OR LIBRARY FUNCTIONS SUCH AS SQR, ATN, RND, CHR$, LEFT$ AND INT ETC. THE PROCEDURES FOR THESE ARE IMBEDDED IN THE ROM AS BASIC UTILITIES. STEVE OLNEY'S EXTENDED BASIC "WAKES UP" A FEW MORE, SUCH AS DEFINT, CSNG AND STRING$.

UNFORTUNATELY ONE OF THE OMISSIONS FROM THE FULL LEVEL II IMPLEMENTATION ON THE VZ IS THAT THE USER DEFINED FUNCTIONS (DEF FNX) ARE NOT SUPPORTED IN ANY WAY. RUSSELL HARRISON'S EXTENDED BASIC XB DOES PROVIDE SUPPORT FOR DEF FN HOWEVER. NOTE THAT FUNCTIONS CAN ONLY RETURN A SINGLE VALUE TO THE CALLING PROGRAM.

THE LACK OF THIS FEATURE OFTEN CROPS UP WHEN ATTEMPTING TO CONVERT PROGRAMS TO RUN ON THE VZ - BUT WRITTEN IN OTHER DIALECTS OF BASIC. THE CONCISE CODING INHERENT IN FUNCTION STATEMENTS IS ALSO A DESIRABLE FEATURE WHEN WRITING CLEAR CODE. FORTUNATELY A FAIRLY SIMPLE REMEDY IS AT HAND AND DESCRIBED BELOW.

THE FUNCTION STATEMENT HAS TWO COMPONENTS. THE FIRST IS THE DEFINITION OF THE FUNCTION, AND THE SECOND, IS THE ACTUAL IMPLEMENTATION OR CALL TO THAT DEFINITION. LET'S EXPLAIN.... SUPPOSE THAT WE WISH TO FREQUENTLY COMPUTE THE AREA OF A CIRCLE GIVEN A NUMBER OF VALUES FOR THE RADIUS. THE COMMAND LINE

```
10 DEF FNA(X) = 3.1416 * X * X
```

SHOULD BE DECLARED EARLY IN THE PROGRAM. DEF MEANS DEFINE, FNA MEANS FUNCTION A (ANY LETTER FROM A TO Z CAN BE USED.), AND (X) IS A DUMMY ARGUMENT FOR THE RADIUS USED BY THE FUNCTION. THE RIGHT HAND SIDE OF THE ASSIGNMENT IS THE EASILY RECOGNIZED FORMULA FOR CALCULATING THE AREA OF A CIRCLE.

LATER IN THE PROGRAM, THE VARIOUS VALUES FOR THE RADIUS ARE ASSIGNED TO VARIABLE V (EITHER FROM DATA OR INPUT STATEMENTS.) AND THEN THE FOLLOWING LINE MAY APPEAR -

```
210 PRINT V,FNA(V)
```

THE TWO LINES GIVEN (10 AND 210) WILL TOGETHER, PRINTOUT THE RADIUS AND THE CORRESPONDING AREA FOR A CIRCLE. THE CALL BY FNA SENDS THE INTERPETER OFF THROUGH THE BASIC PROGRAM TO FIND THE DEF FNA STATEMENT. WHEN LOCATED, THE VALUE V IS SUBSITUTUED FOR THE DUMMY VARIABLE, THE CALCULATION CARRIED OUT, AND THE RESULT RETURNED TO THE CALLING SEQUENCE. (ONLY A SINGLE VALUE CAN BE RETURNED.)

AS ALREADY STATED, THIS NEAT CONSTRUCT DOES NOT EXIST IN VZ BASIC. THE APPROPRIATE USE OF THE GOSUB/RETURN COMMANDS CAN OVERCOME THIS SHORTFALL. THE GOSUB COMMAND CAN RETURN MANY VALUES - BUT A FEW MORE ASSIGNMENTS ARE REQUIRED BEFORE GOING TO THE SUBROUTINE AND YOU HAVE TO KEEP TRACK OF THE SUBROUTINES' LINE NUMBER.

AN EXAMPLE BEST ILLUSTRATES THIS - AND LET'S USE THE PREVIOUS EXAMPLE TO SHOW HOW IT CAN BE IMPLEMENTED ON THE VZ.....

```
10 INPUT "ENTER RADIUS OF CIRCLE ";R
20 GOSUB 1000
30 PRINT"RADIUS ";R,"AREA ";A,"CIRCUMFERENCE ";C
40 GOTO 10
1000 A=3.1416*R*R
1010 C=2*3.1416*R
1020 RETURN
2000 END
```

NOT TOO DIFFICULT TO SET UP IS IT? NOTICE THAT TWO VALUES (A AND C) WERE RETURNED. THE CODING AND PROGRAM FLOW IS NOT QUITE AS CLEAR AS USING THE FUNCTION STATEMENT.

HAVE FUN! AND DON'T BE FOXED BY FUNCTIONS WHEN NEXT CONVERTING PROGRAMS ONTO THE VZ.

TECHNICAL-DATA SHEET # 1

WORDPRO & DD POWER SUPPLY CIRCUITS:

BOTH CIRCUITS WERE SUPPLIED BY DAVE BOYCE SOME YEARS BACK AND ARE PUBLISHED AS A SUPPLEMENT TO THE TECHNICAL REFERENCE MANUAL.

WORDPRO CIRCUIT: - CAN SOMEONE HELP WITH A TRUTH TABLE FOR THE DECODING CIRCUIT PLEASE AS IT HAS SOME UNUSUAL DECODING, ED.

DD POWER SUPPLY CIRCUIT:

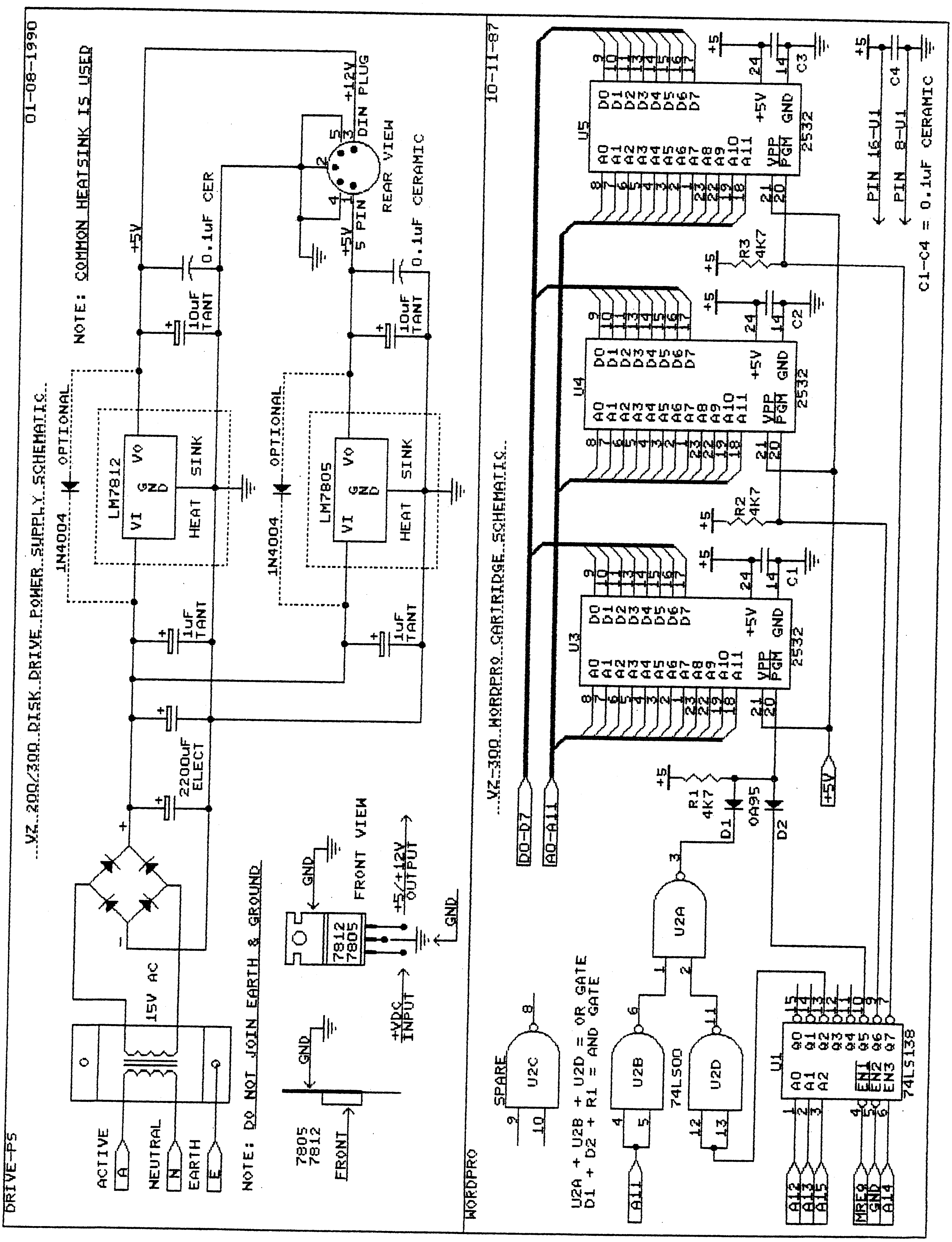
THIS IS A STANDARD POWER SUPPLY CIRCUIT WITH NOTHING FANCY. IT'LL WILL BE INVALUABLE FOR THOSE WITH A CROOK POWER SUPPLY UNIT OR FOR THOSE WHO NEED ONE AND WOULD LIKE TO BUILD ONE THEMSELVES NOW THAT THEY ARE NO LONGER AVAILABLE FROM DICK SMITH.

NOTE: IF THE DEMAND IS THERE I COULD PREPARE A PROJECT FOR PUBLICATION, BUT PLEASE LET ME KNOW AS SOON AS POSSIBLE, ED.

TECHNICAL DATA SHEET # 1

WORDPRO TAPE/DISK WORD PROC. CARTRIDGE CIRCUIT

DISK DRIVE POWER SUPPLY CIRCUIT



DRIVE HEAD PARK UPDATE

As a two drive user I found if I used PARK in DRIVE 2 it would PARK DRIVE 2. On power up or reset DRIVE 1 would bang its head off. Jason Oakley modified it so PARK would always PARK DRIVE 1 regardless of which drive it was run from.

Load PARK source code into your Editor Assembler and insert the following lines by typing I19 at the command prompt.

```
020 ;SELECT DRIVE 1
021       LD   A,10H
022       LD   (IY+0BH),A
```

As you INSERT the lines 020-022 the old lines from 020-070 are being renumbered to 023-073 to end of program. Next EDIT lines 063 and 072 as shown below.

```
063 *      PARK1 BY DAVE MITCHELL
072 *      DRIVE1 HEAD IS PARKED*
```

That finishes the modifications. Save the source and object codes as shown below or according to the SYNTAX of your Editor Assembler.

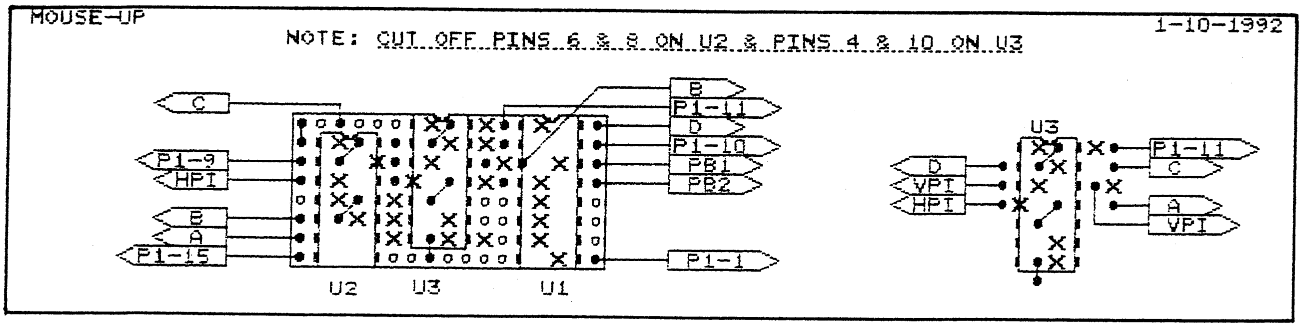
```
O7200H     - Set Origin
A          - Assemble
TS:PARK-1  - SAVE source code
TO:PARK1   - SAVE object code
```

Regardless of which DRIVE you BRUN"PARK1" from, DRIVE 1 will always be the DRIVE which is PARKed, which means no more un-intentional head banging. Happy PARKing !!!

GARY BULLEY VZMOUSE ERRATTA:

Some errors crept in when I was preparing Gary's VZMOUSE project in last issue. My apologies to Gary Bulley and all concerned, Ed.

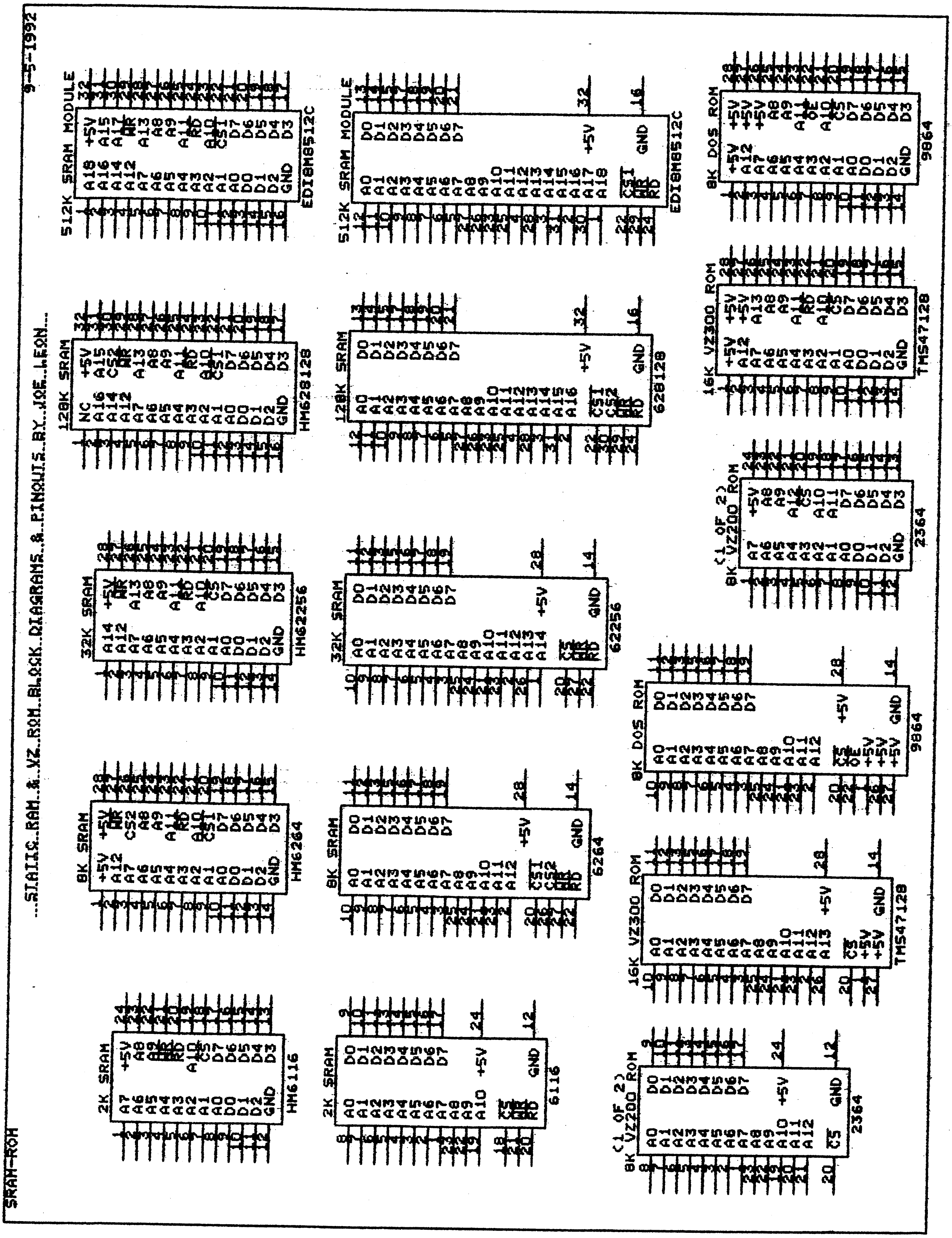
The errors were in the vero board layout and connections. The corrected (I hope) version is shown below. Do not forget to cut off pins 6 & 8 on U2 and pins 4 & 10 on U3 which are not used as it simplifies board layout. The 'B' connection from pin 5 on U2 is made directly to pin 3 on U1. Other connections are as shown.



TECHNICAL DATA SHEET # 2

STATIC RAMS & VZ ROMS

PINOUTS AND BLOCK DIAGRAMS ARE PRESENTED OF STATIC RAMS FROM 2K TO 128K, 512K MODULE, VZ200/300 8K & 16K BASIC ROMS AND 8K DOS ROM.



*** DAVE MITCHELL SOFTWARE ***

E & F WP PATCH 3.3: PATCH 3.3 WRITTEN BY DAVE MITCHELL WILL CONVERT YOUR E & F TAPE WORD PROCESSOR FOR FULL DISK USE WHILE RETAINING ALL ORIGINAL FUNCTIONS. IT ALSO HAS SHIFT LOCK AND PRINTER CONTROL CODES WHICH CAN BE IMBEDDED IN TEXT AND SAVED TO TAPE OR DISK.

BSTWP.F: THIS UTILITY PROVIDED WITH PATCH 3.3 WILL CONVERT BASIC PROGRAMS AND ED/ASS. SOURCE CODE FILES INTO WORD PROCESSOR FILES.

PRICE: AUS/NZ AU$20.00 - UPDATE - AUS-$10.00 - NZ-AU$11.00.

EXTENDED DOS V1.3: THESE COMMANDS ARE AT YOUR DISPOSAL: MERGE, DIRA, DIRA, DIRB, LDIRB, OLD, OLD., DEC, HEX, MENU, CODE, LTAB, MOVE AND UPDATE, STATUSA AND LSTATUSA. STATUSA AND LSTATUSA ALSO WORKS WITH VERSION 1.0 DOS

PRICE: AU$15.00 - POSTAGE INCLUDED

MENU/FILE COPIER: THIS UTILITY WILL READ YOUR DISK DIRECTORY AND PRESENT YOU WITH SEVERAL OPTIONS. USING THE CURSOR YOU CAN RUN/BRUN ANY PROGRAM OR SELECT FILE COPY, REN, ERASE, DRIVE 1 OR 2, ETC. BESIDES COPYING TEXT AND BINARY FILES ALL OTHER FILES CAN BE COPIED AS WELL EXEPT FOR DATA FILES.

PRICE: AU$15.00 - POSTAGE INCLUDED

FOR PURCHASE OR INFORMATION CONTACT:

DAVE MITCHELL 24 ELPHINSTONE STREET
NORTH ROCKHAMPTON QUEENSLAND 4701
AUSTRALIA - PHONE: (079) 27 8519

*** PETER HICKMAN SOFTWARE ***

VZ DISASSEMBLER: WHAT, ANOTHER DISASSEMBLER? BUT, YOU HAVE ALREADY GOT ONE? THIS ONE IS DIFFERENT! THIS PROGRAM IS ENTIRELY WRITTEN IN MACHINE CODE. IT ACTUALLY RUNS ABOUT 40 TIMES FASTER THAN D.S.E.'S DISASSEMBLER (OR ANY ONE ELSE'S). IT WILL DISASSEMBLE ANY PROGRAM THAT YOU CAN BLOAD INTO MEMORY. IT WORKS WITH ANY VZ CONFIGURATION. IT DISASSEMBLES EVEN THE 88 EXTRA Z80 OPCODES THAT ZILOG DOESN'T ADMIT TO.

PRICE: AU$25.00 - PRICE INCLUDES HARDCOPY MANUAL.
TAPE AND DISK VERIONS AVAILABLE.

VZ MODEM SOFWARE: DID YOU WANT TO TALK TO OTHER COMPUTERS VIA A MODEM? DID YOU BUY THE DSE TERMINAL EPROM, ONLY TO DISCOVER THAT IT ONLY WORKS WITH TAPE. IT ONLY ALLOWS YOU TO PRINT FILES, NOT SAVE THEM OR SEND THEM!

YOUR PROBLEMS ARE SOLVED! THE HICKMAN BROTHERS, PETER AND ANDREW, HAVE A BRAND NEW PROJECT WHICH WILL ALLOW YOU TO SEND, RECEIVE & SAVE FILES VIA A MODEM. IT WORKS WITH DISK!

SALE PRICE: $25.00 - INCLUDED ARE INSTRUCTIONS FOR THE HARDWARE MODIFICATIONS. A SMALL MODIFICATION IS NEEDED TO YOUR DISK CONTROLLER. YOUR USER GROUP MAY HELP YOU MODIFY YOUR COMPUTER TO USE THIS EXCITING NEW SOFTWARE!

THE MANUAL IS SUPPLIED ON DISK FOR PRINTING OUT WITH YOUR DISK VERSION OF E & F W/PROCESSOR. IF YOU DO NOT OWN AN E & F W/PROCESSOR THEN PLEASE ENCLOSE ANOTHER $5.00 (TOTAL $30.00) FOR PHOTOCOPYING AND POSTAGE OF THE MANUAL.

FOR PURCHASE OR INFORMATION CONTACT:
PETER HICKMAN PO BOX 8 WERRINGTON 2747

** CONTRIBUTIONS TO THE JOURNAL **

IF YOU ARE THINKING OF CONTRIBUTING TO THE JOURNAL THE PREFERED FORMAT IS BASIC LISTINGS, WORD PROCESSOR OR SOURCE CODE FILES ON TAPE OR DISK. FILES FROM THE FOLLOWING WORD PROCESSORS CAN BE ACCEPTED :-

E & F TAPE OR DISK PATCH 3.1-3.3, WORDPRO CARTRIDGE, WORDPRO PATCH, MOST SOURCE CODE FILES AND ALL QUICKWRITE WORD PROCESSOR FILES.

** CLUB MEETINGS — ALL WELCOME **

FIRST FRIDAY OF MONTH

** FUTURE MEETINGS — NEW VENUE **

AS MENTIONED BEFORE WE NO LONGER MEET AT JNC, BUT AT VARIOUS MEMBERS HOMES. MEETINGS WILL BE ONCE A MONTH AS BEFORE WITH THE DATES BEING FIRST FRIDAY OF THE MONTH.

BECAUSE OF SOME LOCAL MEMBERS HAVING TO WORK SHIFTWORK MEETING DATES WILL BE ADJUSTED TO ACCOMODATE THEM. WHETHER YOU ARE A LOCAL MEMBER, INTRA OR INTERSTATE VISITOR PLEASE CHECK WITH JOE LEON FIRST BEFORE COMING OUT.

JOE LEON 33 TIGHES TCE TIGHES HILL 2297 (049) 692 399

* CLUB COMMITTEE & SUBSCRIPTIONS *

PRESIDENT - ROSS WOODS - SECRETARY/EDITOR - JOE LEON
COMMITTEE MEMBERS - COLIN BRIDGE - PETER JONES

SUBSCRIPTION TO - AUST. - 3 ISSUES $11.00 - 6 ISSUES $21.00
H.V.VZ.JOURNAL - N. Z. - 3 ISSUES $13.00 - 6 ISSUES $26.00

FOR MORE INFORMATION CONTACT:

JOE LEON 33 TIGHES TCE TIGHES HILL 2297 (049) 692 399 AUSTRALIA

NOTE : PRICES INCLUDE POST & PACKING

** VZ USER GROUPS & PUBLICATIONS **

VZ DOWN UNDER - VZ MAGAZINE - 6 ISSUES - $18.00 PER ANUM
HARRY HUGGINS 12 THOMAS SREET MITCHAM VICTORIA 3132

WAVZ - WESTERN AUSTRALIA VZ USER GROUP
GRAEME BYWATER PO BOX 388 MORLEY W A 6062

BRISBANE VZ USERS WORKSHOP - C/O 63 TINGALPA ST. WYNUM WEST 4178
SOFTWARE FOR SALE - DISK MENU

SAPPHIRE PRODUCTIONS - VZ DISK MAGAZINE - PUBLIC DOMAIN
NOTE : VZ DISK MAGAZINE HAS CEASED PRODUCTION

NOTE : WHEN WRITING TO ANY ABOVE OR H.V.VZ. USERS' GROUP FOR INFORMATION PLEASE ENCLOSE A S.S.A.E. OR NZ 2 INT. REPLY COUPONS.